LA-UR -83-3393

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

TITLE HOW TO TALK TO YOUR COMPUTER--LITERALLY

AUTHOR(S) Wendell Ford

SUBMITTED TO 1983 Fall DECUS U. S. Symposium, Las Vegas, Nevada, October 24-28, 1983 Proceedings

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Los Alamos

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

HOW TO TALK TO YOUR COMPUTER--LITERALLY

Wendell Ford
Los Alamos National Laboratory
Los Alamos, NM 87545

ABSTRACT

Voice I/O is a rapidly growing technology. The techniques are still in the formative stage and thus are still somewhat primitive. However, useful tasks can be accomplished within the limitations of today's hardware. No reference to specific hardware will be given, but guidelines for selecting and using hardware will be provided. The guidelines include vocabulary size, method of training, upload/download capabilities, user control of recognition parameters, package form-factor, and information returned to the user. Hints will be given for working with hardware limitations. Some current and proposed uses for voice I/O at Los Alamos National Laboratory will be described. These applications include personnel identity verification, instrument control, data entry, alarm annunciation, and issuing operating system and editor commands.

INTRODUCTION

Many advances in the field of voice I/O for computers have resulted in the availability of relatively low-priced speech recognizers of various forms; however, part of the affordability has been obtained by accepting reduced capability. Machines still cannot carry on a normal conversation such as between two human beings as has been popularized in the science fiction literature. Consequently, expectations from the currently available products have been too high. The amount of software support from the vendors in the form of utilities is very small. This places a burden on the user because some software is needed before any useful application can be achieved. Voice output is much more advanced than voice input: reasonable text-to-speech systems are available. Practical applications exist for voice recognizers if the designer realizes the limitations of current practice and works within them. This paper will concentrate on the use of voice as an input. A brief discussion of voice output (or voice response, as it is often called) is included for completeness. The views expressed are those of the author as a system designer and user of commercially available recognizers. The field is open for rapid advances and a potential user should be aware of the current conditions and make adjustments to take advantage of the latest techniques.

TYPES OF VOICE RECOGNIZERS

There are a number of different ways to classify recognizers; each emphasizes some feature. These features have a considerable effect on how the recognizer may be used. It is important to realize that by most measures the recognizer is not equivalent to a human listener. The human listener makes use of considerably more facilities than the mere processing of the acoustic signal. As artificial techniques are advanced, more and more of the characteristics of humans may be incorporated, but for now most are lacking. A successful designer of a system that uses a speech recognizer will learn as much as possible about the internal workings of the recognizer he is using. Such knowledge can avoid many potential problems and help to work around the limitations as well as making optimum use of the features that are provided.

Current recognizers are not capable of capturing the meaning of a running conversation as is often depicted in science fiction. You will find three terms used to describe what is really done. The first is "isolated word." This means that the utterances must be separated by a definite pause. There are actually two times that are important: the time during which the input must be above a certain level before the recognition process begins, and the length of silence required to indicate the end of an utterance. Related to these are the minimum and maximum lengths that an utterance may have. The term "utterance" is used here because the speech segment to be recognized may be a phrase; it must merely meet the timing requirements discussed above. In that sense the term "isolated word" is a little misleading. It is important to know what action is taken with respect to utterances that are too long. Some devices ignore any utterance that exceeds its maximum length, while others may indicate a rejection of the utterance. The first action allows normal conversation to be interspersed with the recognition task.

The other two classifications are "continuous speech" and "connected speech." These two classifications are not clearly distinguished and clarification should be obtained from the vendor to see what these terms really mean. The latter generally means that no unnatural pauses between words are required, but that the word boundaries must be crisp and distinct. Continuous speech would imply something akin to normal conversation. Often the two terms are used interchangeably. The user should be certain what is allowed and what is not.

Word recognizers may also be classed as speaker independent or speaker dependent. Speaker dependence means that each user must create a set of voice patterns in the machine to represent his voice for each utterance. The patterns are generally called templates, and the process of generating them as "training" the recognizer (not to be confused with training the user to use the machine). Speaker independence implies that the user need not go through the "training" process, usually because a universal set of templates has been created and stored in the machine by the manufacturer. The vocabulary for speaker independence is very limited, usually just the digits and a few others, and is determined at the time of manufacture.

The equipment available today are acoustic pattern recognizers. The input could be any acoustic signal, not just speech. The recognition algorithms are based only on matching the patterns produced from the input signal to a template or reference pattern. Much research has been done and is still being done on systems with the broader goal of speech understanding. Speech understanding implies
that correct recognition of every word is not necessary, but that context and other sources of knowledge are used to determine the meaning of the entire message. This is much the way humans communicate.

HOW A RECOGNIZER WORKS

A generic recognizer block diagram is shown in Fig. 1. Not every recognizer will have every block exactly as shown, but the function must be present. The acoustic signal must be applied to the input to the recognizer. Usually the input is through a microphone although any audio device could be used such as a radio receiver or tape recorder. The input signal is then amplified to a level consistent with the processing circuitry. The next block can take on a variety of forms depending on the design philosophy of the recognizer. It may be some sort of spectral analysis of the waveform or extraction of some other features. The analysis may be performed by either analog or digital methods. Analog-to-digital conversion is required before or after this block. More and more, this operation is being performed by specialized integrated circuits. The data rate of the processed signal can be very high; for example, 16 parameters sampled every 5 ms and digitized to 8 bits produces 3,200 8-bit bytes per second of speech. Because speech is a relatively slow phenomenon, and the speech signal contains considerable redundancy, the data can be compressed to perhaps as few as 100 bytes per second of speech.

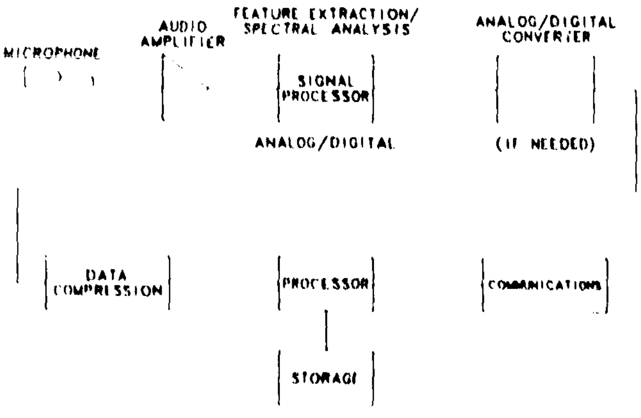

Fig. 1. Generalized Block Diagram of a voice recognizer.

There is a control element that is usually a microprocessor. The controller determines if the input signal is a template to be stored or an unknown to be checked. If it is a template, the patterns are stored and indexed. If the input is unknown, the signal is compared against the reference patterns of interest. If a suitable match is found, information about the matching pattern is made available; if not, that information is also conveyed. Storage for the reference patterns is provided: RAM for speaker dependent systems or ROM for speaker independent ones. Communication circuitry allows operation over particular buses or through serial or parallel ports.

SPECIFICATIONS--WHAT DO THEY MEAN

In evaluating a recognizer for an application, the designer must deal with the various specifications. The usual first question is the general "How good is it?" What is usually meant is "What is the recognition accuracy?" There are many published values for recognition accuracy, most often in the range of 98% to >99%. The figures should be approached like the EPA gasoline mileage figures for automobiles. That is, the figures are certainly attainable under a given set of conditions, but you may not achieve them in your application. In any case, the recognition accuracy is sufficient for a large number of applications and is generally not a problem except possibly for some very low-grade devices. By taking the proper precautions as given in a later section, you can achieve the accuracy needed for most applications. Early system design should assume less than perfect operation and be prepared to deal with it.

Often the most important specification is the vocabulary size. This is the number of reference templates that are stored at one time in the recognizer, and is thus the maximum utterances that can be compared with the input signal at one time. Vocabulary size is determined by the amount of memory set aside for template storage. Vocabulary size may not be a real limitation if adequate provision is made for shifting templates in and out, and the system designer uses it prudently as is discussed in a later section.

Other important specifications have to do with timing considerations. The response time (time to make a decision) is a function of the size of the active vocabulary. It may also depend on template size. Response time depends on the total number of byte comparisons that must be made. Rarely will the response time be a limitation in a well designed application, but it should be checked. Other timing considerations are the minimum utterance length, maximum utterance length, and required pause between utterances. The designer should be aware of these in designing the application.

CHARACTERISTICS AND FEATURES

Besides the specifications there are many other features that may be important in selecting a recognizer or designing the application. Perhaps the single most important of these is the capability to upload the reference templates to a host or mass storage device. This ability allows a larger vocabulary and, in the case of speaker dependent systems, many speakers. Closely related is the template size in bytes. The upload/download time is a function of the communications method and the template size. A template of 1000 bytes takes 10 times as long to load as one of 100 bytes. For reference, using round numbers, 1000 bytes per second are transferred at 9600 baud. Unless care is taken, the upload/download times can easily become excessive. If these times are critical, parallel instead of serial communications may be needed, fewer templates may be transferred, direct bus access may be needed, or a recognizer with a smaller template may be selected. The template size also determines the amount of external mass storage needed and may determine the ultimate vocabulary size or number of speakers.

The amount of host overhead should be considered. One factor affecting the overhead is the nature of the information given by the recognizer. When an utterance is recognized, only a few fixed ASCII characters may be sent or an arbitrary ASCII string. Operation of the recognizer can be independent or it can be host driven; sometimes a given recognizer can be either. There may be an advantage in certain applications in a very interactive host.

Template generation methods may vary such that different features of an utt rance are important. The better the designer understands the generation method, the better the chance his application will succeed. The structure and content of the vocabulary may be determined by the way the template is generated. Some questions to be asked include:

- How many training passes are needed?
- Can the user determine the number?
- Is each utterance repeated or the list repeated in sequence?
- Is retraining all or part of the vocabulary possible?

Often many of the recognizer parameters are adjustable by the user and may be tailored for specifi applications. The most common of these is the threshold, that is, the level of match between the input signal and a reference template, which must be exceeded. Another is the distance between the template closest to the input signal and the ext closest. If such adjustments are to be made, he designer must understand their effects or performance may actually be degraded. Other examples of such parameters are the sound level required to signify the beginning of an utterance or the level signifying an end of an utterance.

The length of the intraword silence also plays a part; for example, there is a pause in the word "eight" before the final "t", or the "t" may not be sounded at all.

Another feature important in selecting a recognizer is its intended operating environment. A board level product may have to plug into a host bus (CPU or terminal). Some box level products require an external host (or perhaps just a controller, like a terminal) and others are completely stand-alone.

WHEN TO USE VOICE INPUT

Voice input is not appropriate for all applications; the designer should take care to see that it is appropriately applied. There should be a distinct and recognizable advantage to the use of voice input. Some characteristics of an appropriate application are easily identified. In cases where the hands and eyes are busy with some other task, voice input can speed up the task as well as reduce possible errors. Examples of this occur in inspection processes where the operator is physically handling the material being inspected and the inspection process must be interrupted to enter data. Another use is where a single spoken word can replace a number of keystrokes, much as user-assignable function keys on a terminal are used. The ASCII string sent upon an utterance recognition can literally be anything (subject to a maximum number of characters).

A degree of operator mobility may be achieved by using voice input, either through a wired microphone or over a radio link (wireless microphone). This allows data to be entered at the point of collection and without a delay to go to a data entry station. The requirement for the audio input device must not place any restriction on operating procedures. In cases when the data (or control command) entry point is a well-defined location, fixed microphones may perform better than other data entry devices. Control over some device may often be more safely and conveniently exercised by voice input, especially when the device is remote and requires interruption of the operator's normal routine.

There are applications when voice input may not be the best. A good typist is fast and merely replacing each keystroke with a spoken word would be painfully slow. A single spoken word replacing an entire phrase or more might be more efficient, as when a single word could enter a company name and address. Some clever combination of voice and keyboard input could greatly enhance a skilled operator's efficiency Sometimes the data to be entered are sensitive and should not be spoken aloud (your password?). An environment with a lot of transient noise or with frequent turnover of personnel is not conducive to voice input. To be used effectively, the vocabulary should be well defined and limited just as strictly as syntax is limited and controlled in any other computer interaction. This does not rule out tailoring a vocabulary for each operator, but rather requires that it must be strictly adhered to. Consistency is the key to successful voice input In some cases there may be objections to the use of a microphone for safety or cosmetic reasons, particularly if a head-worn microphone is to be used.

Finally, voice input may not be the appropriate device for the application. A bar-code reader is generally faster than typing in digits or speaking them. A person's voice tends to change under stressful conditions; therefore, voice input is usually not successful under conditions of stress. Temporary voice changing conditions such as colds should be expected, and ways of handling them included in the system design, such as manual override or operator intervention.

APPLICATIONS FOR INPUT

One of the most successful applications of voice input to date has been in the area of quality control inspection where the data rate is low, but entering by other means interrupts the worker's task. It has been applied in the manufacture of printed circuit boards, appliances, and custom hybrid circuits.

A person's voice is a unique characteristic that is always with the person. It has often been looked at as a means of uniquely identifying someone. Voice prints caught the imagination of the press and public a few years ago. The promise was never fulfilled because of many practical problems including a great deal of skill needed by the analyzer. Voice prints are essentially plots of time vs frequency, with energy represented as a third dimension by the density of the plot for a particular utterance. This is different than the templates used in today's word recognizers. However, because the recognizers are speaker dependent, some speaker discrimination is possible. The task of speaker verification has been attempted at Los Alamos; that is, the person makes an initial claim of identity through a badge reader, entering a password or ID number, and the system either verifies the identity or rejects it (Fig. 2). This is a much easier task than identification because the input is compared with only one set of templates, that is, the claimed identity. Upon input of the ID number the voice templates associated with the number are downloaded into the recognizer. Then the user is prompted to repeat an utterance. The incoming utterance is checked against the template and a decision made. A series of randomly selected prompts is used and the entry/no entry decision is based on the overall result. A total of 16 different utterances is used with the vocabulary carefully selected for best discrimination. Various strategies of number of utterances recognized vs those not recognized have been investigated. A high-quality voice synthesizer is used to prompt the user during both the training phase (template generation) and in actual use because it was found that the user would speak much more consistently. So far, this system has only been used experimentally within the laboratory and demonstrated during an "open house." Under these limited conditions, the results showed an impostor success rate of about 1 in 400 when the valid user rejection rate was about 1 in 10. There is a tradeoff between the two types of errors. The impostor success rate is a worst case because it implies that the impostor has the luxury of being able to make repeated trials against the same template, or that he is able to try all templates after he has obtained the primary means of identification. Administrative procedures could be used to eliminate unrestrained trials. A preliminary discussion of this work was reported in Ref. 1.

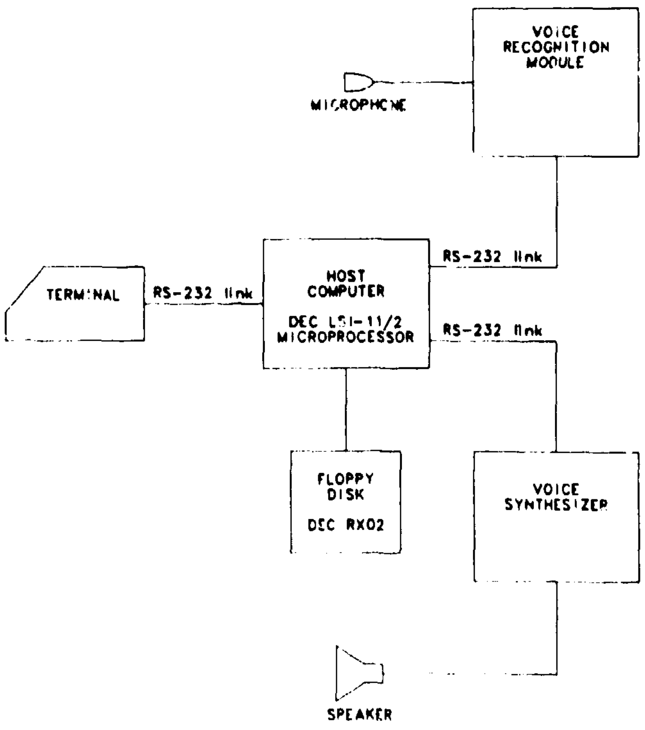

Fig. 2. Block diagram of a stand-alone speaker verification system. Other versions interface to a host security system.

Work on hazardous materials is often done in enclosed stainless steel boxes, with the operator working through heavy rubber gloves permanently mounted in the side of the box. A process line may consist of a long line of interconnected glove boxes. Measurements of certain quantities may be required for process control or other purposes. Measurement instruments may be quite sophisticated and require operator attention during the measurement. It can be very disruptive for the operator to leave his task to attend to an instrument. For example, a calibration run may be required before actual data are acquired. By controlling such an instrument by voice, the task is not interrupted nor is attention diverted. We have built one such control for an instrument as is shown in Fig. 3. All operation is controlled through voice commands.

An example of voice entry that illustrates many of the features is the interface to a computer operating system. We do a lot of design that uses LSI11's running RT11. The program development is done on a PDP11 and transferred to the LSI11 via floppy disks. RT11 has a little over 50 monitor commands. This is well within the 100-word vocabulary size of many recognizers. Each monitor command is called up by voice. When the command is accepted, a new vocabulary that consists of the options valid for that command is downloaded into the recognizer, and the proper command string is

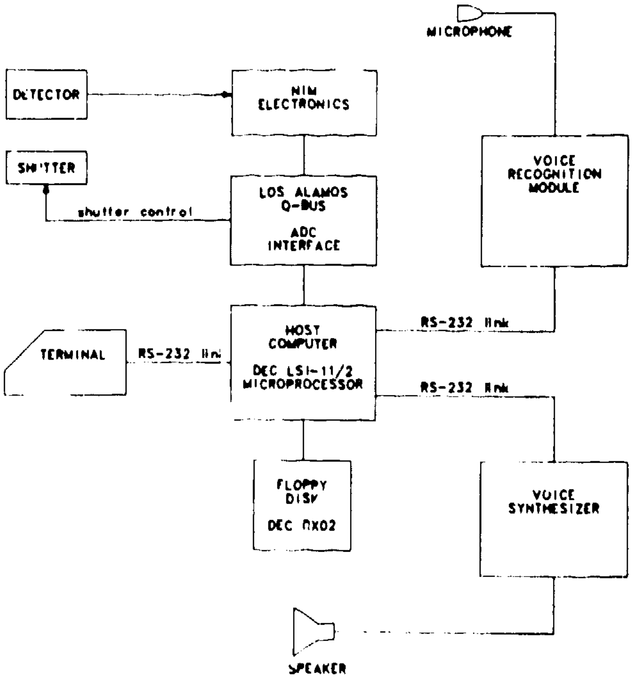

Fig. 3. Example of a voice controlled instrument. The computer system was already required by the instrument so voice control only added two circuit boards in the computer backplane and a small amount of software.

sent to the computer. When the option is recognized, the correct string is sent and templates for valid arguments are loaded; these usually consist of the current files. The last string sent terminates in a carriage return to initialize command execution. Table I shows representative commands with options and arguments. The advantage is that it is not necessary to remember the exact syntax expected for each monitor command. The right syntax is built in, as is the ASCII string that is sent out in response to each voice input. An example of different syntax is that for a DIRECTORY output to go to the console, the option is TERMINAL, while for a FORTRAN/LIST, the option is :TT:. Under either command the spoken word "terminal" sends the correct string.

Another application is using voice for the keypad editor commands while entries and changes are made via the keyboard. A little practice makes this a very fast and easy-to-use process. Some careful thought is needed to produce a practical system. Voice replacement of a single keystroke might make sense (for example, to go down one line) because the fingers would not have to be replaced on the keyboard, but would not make sense for many repeats of the same key. Both methods can be active at the same time and the most appropriate one used. Control of other utilities may likewise be useful.

MAKING IT WORK

Attention to detail is necessary to produce a successful application. This section gives do's and don'ts to ensure success. The most important thing is not to exaggerate expectations. The capability of voice recognition is limited and the system must be designed within the limitations. Some development time must be allotted; as purchased, the recognizer is not ready to be plugged in and begin doing useful tasks. When one purchases a computer system, it is necessary to have supporting software such as operating systems, compilers, etc., before it can be used. The same is true (although to a lesser degree) for recognizers. Unfortunately, the ready-to-use software packages do not exist and the user must write them. Many have bought a recognizer with the expectation of merely plugging it in and immediately producing a useful function. When that did not occur, they concluded that speech recognition did not work.

The first application for voice input should be chosen very carefully. The application should be real and not just a novelty. Be certain that the user of the system gains something by its use; it should not make his task more difficult or appear to be in the way. The users should be sympathetic to the possible benefits and also to the limitations. If someone does not want a speech input

TABLE I

AN EXAMPLE OF SYNTAX NODES FOR INTERFACING TO THE RT11 OPERATING SYSTEM. ONLY THE VOCABULARY OF ANY ONE COLUMN IS ACTIVE AT ANY GIVEN TIME. RESPONSE TO ANY INPUT AUTOMATICALLY ENABLES THE NEXT LIST OF VALID INPUTS.

SYNTAX NODES (ACTIVE VOCABULARY)

MONITOR COMMANDS	OPTIONS	ARGUMENTS
COPY	BOOT DEVICE SYSTEM ⋮	FILE SPECS ⋮
DIRECTORY	ALPHABETIZE PRINTER SUMMARY ⋮	DEVICE ⋮
FORTRAN	LIST WARNINGS ⋮	FILE SPEC ⋮
LINK	MAP RUN ⋮	FILE SPEC[1] ⋮
EDIT	CREATE KED ⋮	FILE SPEC ⋮

[1] EXAMPLE: MAIN, SUB1, SUB2, ...

system to work, it is very easy to achieve that goal. The user should be thoroughly educated in the advantages and uses of voice input; experience has shown that their training is much more effective if they are not left to do it on their own. The users should be brought in early in the design phase and not have the system forced on them. Emphasize that the important thing is consistency in speech; frustration usually leads to a different form of speech, which causes the system to perform even worse, which leads to more frustration. Tacked-on systems do not have as much chance of success as when speech is included from the beginning. If a redesign of the system is needed, do so, but make certain that something is gained in the process. Casual users will generally not be as successful as those who use the system continually. Practice improves the proficiency of the user.

With real-time uses, the input is through a microphone. The system will work no better than the audio signal given it. The best microphone is a head worn, close-talking, noise-canceling unit; however, many people object to wearing one. Other microphones can be made to work with very little loss in performance if strict attention to placement is paid. Again, consistency is the goal. An easily used on/off switch on the microphone is a necessity under some conditions. An adverse acoustical environment also presents challenges to the system designer.

The system will work no better than the quality of the reference templates will allow. Care in creating the templates will pay great dividends. Follow the manufacturer's recommendations and experiment with variations before the users are introduced to the system. Train the users thoroughly in the procedures to be followed and do not leave them on their own. The templates should be created in the actual environment or as close to it as practical. Make a trial pass after the templates are created and immediately retrain any problem areas. Often a first-time user will not produce a good set of templates. Do not hesitate to redo the templates when performance degrades, especially be prepared (and have the user expect) to retrain during the initial phases of use. The frequency of retraining that is necessary will decrease with the experience of the user.

In designing the system, only a fraction of the vocabulary is needed at any given time. Take advantage of the syntaxing feature, which limits the active vocabulary, even if all templates can be loaded in the recognizer. It is usually possible to have certain key control words active at all times, even when using syntax, but keep their number to a bare minimum. Some appropriate form of feedback to the user is necessary; he needs to know that the word was recognized or not recognized. The form of feedback can be as simple as a beep, print out on a terminal, or something more elaborate. Use what makes sense in the application. Expect that the recognizer is going to make occasional errors and be prepared to deal with them. The type of precautions that are needed will depend on the severity of the consequences of the error. Terminal manufacturers put DELETE keys on their terminals with no apo [illegible] errors are expected; the same is true of a voice input system.

VOCABULARY SELECTION

Selection of the vocabulary is very important. It should be meaningful to the task and to the user. User input in selecting the vocabulary is helpful. Avoid similar sounding or rhyming words. The straight alphabet as it is usually recited is a very poor choice. The phonetic alphabet is better but there are potential problem areas (Ref. 2). Different opinions exist on the optimum length of words. A general rule probably does not exist. The algorithm used in the recognizer may provide some guidance. If the recognizer reduces the data to 16 time slots, for example, it would actually reduce performance to include words (or phrases) that contain more than 16 acoustical events, because dissimilar events would tend to be averaged. If the utterance is too short, full advantage is not taken of the information that can be processed. Our experience has shown that two to four syllables are best, depending on the actual word. Most of the variation in the way words are spoken occurs in the vowels. In the speaker-verification task we look for words with prominent and differing vowel sounds. Speaker-independent recognition would dictate the opposite approach.

The recognition process is triggered when the input signal exceeds some specified level and ends when it is below another level for a certain length of time. Therefore, words that have distinct and sharp initialization and cutoff tend to provide the best performance. There are exceptions, so this rule must be used carefully. Keeping the vocabulary to the minimum necessary to accomplish the task will have many advantages: the recognizer response time is reduced, the possible misrecognitions are reduced, the user has fewer words to remember, training time is reduced, template storage is reduced, download and upload times are reduced, and host overhead is usually lower.

VOICE OUTPUT

The current state of voice output is such that there are no technical or economical reasons for not using it. However, the designer should have a good reason for using it; it should not be just gadgetry. The voice output should offer some advantage over other methods of conveying the message. It may, in fact, supplement or enhance other methods. Auditory signals can reach a number of people scattered over a large area. Bells, horns, buzzers, etc., do the same thing but the information content is limited. A spoken message can provide much more detail. An alarm can sound and instructions can be given for the proper response to the alarm. Spoken messages are also useful when the message recipient cannot receive the message by other means; that is, their eyes are busy with some other task. Indeed voice communication makes it possible for the person to continue with some other task. Situations that require sending messages over the telephone are ideal candidates for voice response.

A wide variety of voice synthesizers are available. Some are better suited for particular applications than others. Some are more easily understood than others. The voice quality that one needs for any task is probably available. However, matching the synthesizer to the application requires some careful consideration. To merely record voice digitally and then play it back with good fidelity requires considerable memory. If voice frequencies of 5KHz are to be preserved, the signal must be sampled at 10KHz or above. Digitizing each sample to 10 bits means that 100K-bits are required for each second of speech. The thrust of the work in voice synthesizers has been to reduce the bit-rate and still achieve reasonable voice quality. Various schemes exist for compressing the data rate (see Table II). Usually there is a tradeoff between data rate and voice quality. Two major methods of voice synthesis are used with many variations. The first method records actual voices and then reduces the data by use of algorithms or coding techniques. The vocabulary has to be predetermined and processed for use by the synthesizer. If properly done, the voice quality can be very high. Individual words or entire phrases may be processed. The individual elements may then be combined in different ways to produce different messages. One should realize that words spoken in isolation sound different than words spoken in continuous speech. The coarticulation causes different inflections and emphasis depending on the context. If stringing the individual words together does not produce good quality, then the desired phrase should be produced as a unit. This first method reduces flexibility and may cost more. The second method of voice synthesis is to synthesize the individual sounds that make up speech and string them together to form words. The elemental sounds are called phonemes; English contains 44 or so of them. Each one may have many different variations and inflections. Current text-to-speech synthesizers usually use this method. In general, the voice quality is lower, but the data rate can be very low (perhaps under a 100 bits per second of speech). The vocabulary may be unlimited. The quality can be improved somewhat by adjusting the individual phonemes.

TABLE II

REPRESENTATIVE DATA RATES REQUIRED FOR VARIOUS VOICE DIGITAL RECORDING TECHNIQUES AND VOICE SYNTHESIZERS

DIGITAL RECORDING	
• DIRECT DIGITIZATION	100,000 BITS/SEC
• LOG PULSE CODE MODULATION	50,000 BITS/SEC
• CONSTANT SLOPE DELTA MODULATION	25,000 BITS/SEC
• CONTINUOUSLY VARIABLE SLOPE (CVSD)	15,000 BITS/SEC
• MOZER ENCODING	2-6,000 BITS/SEC
SYNTHESIS BY ANALYSIS	
• LINEAR PREDICTIVE CODING (LPC)	1-4,000 BITS/SEC
• FORMANT SYNTHESIS	500-1,000 BITS/SEC
SYNTHESIS BY RULE	
• PHONEME SYNTHESIS	100 BITS/SEC

It should be mentioned that the synthesizers are general devices and that sounds other than speech can be produced. The same circuitry could produce speech, music, and other sound effects.

APPLICATIONS FOR VOICE OUTPUT

The current state-of-the-art in voice output is such that a designer should seriously consider using it in addition to or instead of practically any other type communication with a user of his system. The reliability improvement over tape loops and other similar devices is very great. Many work environments have a variety of audio signals to convey messages to their workers. These range from the proverbial five-o'clock whistle to safety and fire alarms. Workers must receive instruction as to the meaning of these various signals. A voice message that would explain the reason for the signal can easily and cheaply be added to any signal system. This is especially true for a visitor or casual worker and would save considerably on training time. Repetitive announcements, either over a PA system or a telephone, can be done with voice synthesis more easily and cheaply than about any competitive system, for example tape loops. It also makes certain types of information readily available over a telephone. Important information can be given by combining the voice synthesis with an autodial system so that a malfunction can be immediately and easily transmitted to the interested party.

CONCLUSIONS

Voice input/output technology as it exists is usable today. Voice output is more advanced than voice input and practically any application requiring voice output can be met. Voice input requires more work on the part of the system designer. He should be prepared to expend some effort on software to interact in a meaningful way with the recognition device. Voice input or output should be considered from the beginning of system design and not something that is added as an afterthought. Not every application is appropriate for voice input at present. It is crucial to choose the application carefully. The user has the ability to make voice input work or not work and should be consulted early in the design phase.

The technology can be expected to improve with time. The rate of progress will depend on the actual and perceived demand. A body of serious users (translate: buyers) is needed as a driving force. The user base will build as more software tools become available. A set of software building blocks is required that can easily be tailored to specific needs such as is available for computer systems today. This audience has the skills necessary to develop the building blocks. I suspect the tasks also exist.

REFERENCES:

1. Wendell Ford and Donald G. Shirk, "Applications of Voice I/O," Nuclear Materials Management X, 432-437 (1982).

2. Michael Pfauth and William M. Fisher, "Voice Recognition Enters the Control Room," Control Engineering, 147-150, (Sept. 1983)